Модел. и анализ информ. систем. Т. **19**, № **5** (2012) 142–151

УДК 004.7.057.4

TCP TIPS: транспортный протокол для ненадежных сетей, передающих чувствительные к задержкам данные

Сивов А.А.[1]

Ярославский государственный университет им. П.Г. Демидова

e-mail: mm05@mail.ru

получена 22 ноября 2012

Ключевые слова: протокол TCP TIPS, транспортные протоколы, TCP, борьба с перегрузкой, сетевое моделирование, ns-2

Описывается протокол TCP TIPS, направленный на эффективное использование доступной пропускной способности сети. Этот протокол реализует проактивную схему борьбы с перегрузкой в сети. Использованная схема позволяет TCP TIPS уступать требуемую долю пропускной способности сети потокам, передающим данные реального времени. При этом TCP TIPS продолжает использовать освобождаемую долю после завершения указанных потоков.

Транспортный уровень занимает ключевое место в иерархии протоколов передачи данных. Он обеспечивает обмен данными между пользователями транспортных услуг (реализующими протоколы прикладного уровня), обладающий необходимыми пользователю характеристиками, такими как надежная доставка данных и их правильный порядок. Наибольшее распространение среди транспортных протоколов, обеспечивающих гарантию доставки данных, является TCP. Помимо надежной доставки и восстановления порядка полученных данных, TCP реализует алгоритм управления потоком данных, цель которого – достичь максимально возможной скорости передачи данных, не вызывающей перегрузку в сети.

Традиционный алгоритм, используемый TCP [1], относится к реактивным схемам избежания перегрузки. Индикатором перегрузки в сети в данном алгоритме является обнаружение потери сегмента данных. Такой способ индикации эффективно работает в надежных сетях, имеющих также маршрутизаторы, размер буферов которых близок к BDP. Однако в сетях, использующих ненадежную среду передачи данных (например, беспроводных сетях) или содержащих маршрутизаторы, обладающие буферами чрезмерно большого размера, использование TCP может быть неэффективно или приводить к значительному росту задержек при передаче данных от отправителя получателю [2].

[1]Исследование выполнено при поддержке Министерства образования и науки Российской Федерации, соглашение №14.132.21.1366, и гранта РФФИ №12-07-31173 мол-а.

Для раннего обнаружения перегрузки и борьбы с проблемой роста задержек при передаче данных могут применяться проактивные алгоритмы. Такие алгоритмы используют для индикации перегрузки темпоральные характеристики потока. Наиболее часто в роли данной характеристики выступает RTT. Классическим примером использования проактивного алгоритма борьбы с перегрузкой является TCP Vegas [3].

Однако применение RTT имеет ряд своих недостатков. В частности, нет возможности пользоваться механизмом отложенной отправки подтверждений. Также использующие RTT алгоритмы не в состоянии отличить перегрузку, наблюдающуюся в канале, по которому передаются данные, от перегрузки в канале, по которому передаются подтверждения. Подобная неверная индикация перегрузки приведет к недоиспользованию имеющейся пропускной способности.

Другим слабым местом многих проактивных алгоритмов избежания перегрузки, а также алгоритмов, направленных на оценку доступной пропускной способности, является использование пороговых значений задержки (таких, как BaseRTT (базовое значение RTT)). Неправильный выбор порогового значения может существенно повлиять на работу таких алгоритмов.

Необходимо использовать отличный от названных способ обнаружения перегрузки, чтобы создать проактивный алгоритм борьбы с перегрузкой, который обладает следующими свойствами: позволяет эффективно использовать доступную пропускную способность сети в ненадежных сетях и при частой смене маршрутов; не вызывает рост задержек при передаче данных, связанный с чрезмерным накоплением данных в сетевых буферах маршрутизаторов; лишен недостатков, характерных для решений, основанных на анализе RTT. В качестве индикатора перегрузки может выступать какая-либо темпоральная характеристика потока данных, передаваемых от отправителя получателю.

При отказе от передачи данных всплесками (burst) и использовании определенных интервалов времени между отправкой сегментов (TCP pacing) такой характеристикой может быть значение межсегментного интервала, использующегося отправителем и наблюдаемого получателем. Анализ значений межсегментных интервалов используется в начальной фазе работы протокола ARTCP [4] для первичного обнаружения перегрузки. Протокол TCP TIPS, рассматриваемый в данной статье, использует эту характеристику в качестве единственного индикатора перегрузки.

В целом, TCP TIPS является развитием протокола ARTCP и заимствует ряд его черт, среди которых можно выделить отказ от использования окна перегрузки, порогового значения «медленного старта» и передачи данных всплесками, разделение логики обработки потерь и алгоритма контроля скорости передачи данных. Однако TCP TIPS имеет также ряд ключевых отличий: отказ от использования RTT для обнаружения перегрузки, совершенно другой механизм изменения скорости передачи данных. Первый раздел статьи описывает алгоритмы, лежащие в основе TCP TIPS. Второй раздел рассматривает некоторые особенности реализации протокола.

1. Алгоритмы протокола TCP TIPS

TCP TIPS использует значения межсегментных интервалов на стороне отправителя для контроля скорости передачи данных. Алгоритм управления потоком передачи данных TCP TIPS основан на анализе значений межсегментных интервалов, применяемых отправителем и наблюдаемых получателем.

Для обеспечения возможности анализа значений межсегментных интервалов, измеренных получателем, на стороне отправителя необходимо предоставить получателю возможность корректно вычислять эти значения и реализовать способ их доставки отправителю. Для решения этих задач TCP TIPS использует два параметра — TI и PS, – давших название протоколу (TIPS является конкатенацией имен параметров). Параметр TI (Time Interval, промежуток времени) используется для передачи измеренного получателем значения межсегментного интервала от получателя отправителю вместе с подтверждениями о доставке сегментов. Чтобы получатель имел возможность определить, что принятые им сегменты являются соседними (т. е. отправитель, например, не передавал между ними сегмент с данными, не доставленный по причине потери), отправитель последовательно нумерует все посылаемые им сегменты. Номер сегмента передается с помощью параметра PS (Packet Sequence, последовательность пакета). Значение этого параметра не имеет ничего общего с номером последовательности TCP, т. к. PS означает номер сегмента в передаче. Он увеличивается при каждой отправке сегмента (в том числе при повторной отправке) и никак не связан с передаваемыми в сегменте данными.

Алгоритмы, применяемые TCP TIPS, можно условно разделить на два пункта: алгоритм оценки доступной пропускной способности сети и алгоритм управления потоком данных, использующий первый алгоритм.

Алгоритм оценки доступной пропускной способности сети основан на том, что минимальное значение межсегментного интервала, наблюдаемого получателем, в сети при отсутствии сторонних соединений и ограничения скорости передачи данных на стороне отправителя определяется пропускной способностью сети и подробно описан в [5]. Опишем кратко принцип этого алгоритма.

В случае скорости передачи данных, превышающей пропускную способность сети, пропускную способность при отсутствии других данных, передаваемых по сети, можно определить следующим образом:

$$B = \frac{8 \cdot N}{P}, \tag{1}$$

где B – пропускная способность сети, N – размер отправляемых сегментов в байтах, P – значение межсегментного интервала, наблюдаемого получателем (интервала времени между прибытием последнего бита пакета k и последнего бита пакета k + 1).

Для определения перегрузки и вычисления доступной пропускной способности сети алгоритм пользуется тем, что, как показано в [5], при суммарной скорости передачи данных, превышающей пропускную способность сети (т. е. в момент перегрузки), выполняются следующие отношения:

$$I = \frac{T \cdot x}{a}, \tag{2}$$

$$P = \frac{T \cdot (a + b)}{a}, \tag{3}$$

где I – значение межсегментного интервала, устанавливаемого отправителем, a – скорость передачи данных, устанавливаемая отправителем, T – минимальное возможное значение межсегментного интервала для данной сети (определяется пропускной способностью сети при фиксированном размере передающихся по ней сегментов), x – пропускная способность сети, P – значение межсегментного интервала, наблюдаемого получателем, b – суммарная скорость передачи данных сторонних соединений в сети.

Если на интервал времени, достаточный для получения оценки значения межсегментного интервала, наблюдаемого получателем, установить такую скорость передачи данных a' (значение межсегментного интервала I'), чтобы перегрузка в сети сохранилась, и при этом изменениями суммарной скорости передачи данных и пропускной способности сети в течение рассматриваемого интервала времени можно было бы пренебречь, то можно получить оценку доступной пропускной способности сети:

$$B = a \cdot \left(\frac{(I - P) \cdot (a' - a)}{a' \cdot P' - a \cdot P} - 1 \right), \tag{4}$$

где B – доступная пропускная способность сети, P' – значение межсегментного интервала, наблюдаемого получателем при скорости передачи данных a', P – значение межсегментного интервала, наблюдаемого получателем при скорости передачи данных a, I – значение межсегментного интервала, используемого отправителем для достижения скорости передачи данных a.

Алгоритм управления потоком данных в TCP TIPS, использующий в своей работе полученную оценку доступной пропускной способности сети, подробно описан в [6]. В его работе можно выделить 4 состояния: медленный старт, мультипликативный сброс, компенсация перегрузки и проба/отмена ускорения, переход между которыми показан на рисунке 1.

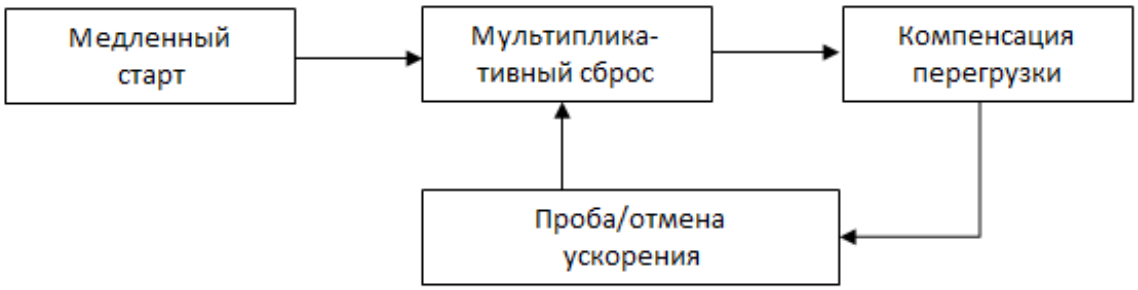

Рис. 1. Взаимодействие режимов работы TCP TIPS

После установки соединения методом тройного рукопожатия TCP TIPS работает в режиме медленного старта. Он устанавливает значение межсегментного интервала равным RTT, вычисленному во время установки соединения. В качестве интервала времени, в течение которого проводится усреднение значений межсегментных интервалов, измеренных получателем, также используется RTT. Пока производится

усреднение значений, отправитель не изменяет скорость передачи данных. Когда усреднение завершено и имеется значение межсегментного интервала, наблюдаемое получателем, TCP TIPS проверяет наличие перегрузки. Считается, что перегрузка имеется, если

$$I < (1 - \varepsilon) \cdot P, \tag{5}$$

где I – значение межсегментного интервала, использованного отправителем для сегментов, усреднение по которым дало значение P для межсегментного интервала, наблюдаемого отправителем, ε – параметр, означающий «зону нечувствительности».

Если неравенство (5) не выполняется, то считается, что перегрузки нет, и TCP TIPS уменьшает значение межсегментного интервала, деля его текущее значение на параметр SSGR > 1. В противном случае TCP TIPS проводит подготовительные действия для получения оценки доступной пропускной способности сети, определяемой с помощью алгоритма, описанного ранее (см. также [5]). С помощью этого алгоритма TCP TIPS вычисляет отношение ak скорости передачи данных к пропускной способности сети и отношение bk суммарной скорости передачи данных сторонних соединений к пропускной способности сети:

$$ak = \frac{\alpha \cdot P' - P}{I \cdot (\alpha - 1)}, \tag{6}$$

$$bk = \frac{\alpha \cdot (P - P')}{I \cdot (\alpha - 1)}, \tag{7}$$

где $\alpha = \frac{I}{I'}$.

После вычисления этих отношений TCP TIPS при наличии возможности избежать перегрузку устанавливает оценочное значение межсегментного интервала I_e:

$$I_e = \frac{I \cdot ak}{1 - bk}. \tag{8}$$

Далее происходит переход к режиму мультипликативного сброса, в котором скорость передачи данных устанавливается заведомо ниже доступной пропускной способности сети (для возможности компенсации перегрузки), производится расчет области компенсации и происходит переход к режиму компенсации.

Значение межсегментного интервала, действующее во время компенсации перегрузки, задается формулой:

$$I_r = \frac{I_e}{MDF} \tag{9}$$

где MDF – множитель мультипликативного сброса, $0 <$ MDF < 1.

Площадь компенсации перегрузки представляет собой сумму площадей прямоугольников длиной RTT и высотой, равной разности скорости передачи данных и оценки доступной пропускной способности. Для вычисления указанной площади можно воспользоваться формулой

$$S = PRTT \cdot \left(P^{-1} - I_e^{-1} + \sum \left(\frac{SSGR^n}{I'} - I_e^{-1} \right) \right), \tag{10}$$

где PRTT – среднее значение RTT, вычисленное в момент обнаружения перегрузки. Первое слагаемое используется в формуле 10, только если выполняется условие $I_e > P$. Остальные слагаемые подставляются в формулу 10, если выполняется условие $\frac{SSGR^n}{I'} \geq \frac{1}{I_e}$.

Длительность работы в режиме компенсации перегрузки вычисляется по формуле:

$$\Delta t = \frac{S}{I_e^{-1} - I_r^{-1}}, \tag{11}$$

где S – величина, определяемая формулой (10).

Выход из режима компенсации происходит в двух случаях: по истечении интервала времени, определенного формулой (11), или при обнаружении перегрузки согласно формуле (5). Перегрузка в режиме компенсации может возникнуть из-за действий других соединений или уменьшения пропускной способности сети по каким-либо причинам (например, смена маршрута для соединения). При выходе из режима компенсации соединение переходит в режим пробы/отмены ускорения. Однако если причиной выхода является перегрузка, TCP TIPS сохраняет эту информацию в состоянии соединения. После выхода из режима компенсации значение межсегментного интервала устанавливается равным I_e, если причиной выхода стало окончание промежутка времени Δt, или равным P, полученным при вычислении среднего арифметического значения поля TI, если причиной выхода является перегрузка.

При наличии перегрузки будет осуществлен мультипликативный сброс. Площадь компенсации в данном случае определяется формулой:

$$S = 2 \cdot SRTT \cdot ((I_r^{-1} - I_e^{-1}) - (P^{-1} - I_e^{-1})), \tag{12}$$

где $SRTT$ – сглаженное значение RTT.

При возникновении перегрузки в процессе работы режима пробы/отмены ускорения площадь компенсации также будет определяться формулой (12), но вместо I_r и P будут использоваться действовавшие на стороне отправителя значения межсегментных интервалов.

Режим пробы/отмены ускорения направлен на проверку наличия большей пропускной способности при отсутствии перегрузки и быструю компенсацию перегрузки, возникшей в результате неверного ускорения передачи данных.

Ускорение и отмена ускорения задаются формулами (13) и (14) соответственно.

$$I_n = min \left\{ I', \frac{I}{1 + \beta(I)} \right\}, \tag{13}$$

$$I_n = \frac{I_o}{1 - \beta(I_o)}, \tag{14}$$

где I_n – устанавливаемое значение межсегментного интервала, I' – текущее значение межсегментного интервала, I – значение межсегментного интервала, для которого установлено наблюдаемое получателем значение межсегментного интервала P, $\beta(I)$ – функция ускорения, I_o – значение межсегментного интервала, соответствующего отсутствию перегрузки в сети.

Функция ускорения определяется формулой (15)

$$\beta(I) = C \cdot I^{3/2}, \tag{15}$$

где C – коэффициент, соответствующий максимальному допустимому приросту скорости. Этот коэффициент можно установить равным оценке пропускной способности сети. Чтобы C соответствовало максимальному приросту скорости, для $I > 1$ следует установить $\beta(I) = C$. Для значений $I < 1$ функция $\beta(I)$, выраженная формулой (15), делает прирост скорости передачи данных обратно пропорциональным квадратному корню скорости, что позволяет уменьшать разность скоростей при совершении двумя каналами ускорения, иметь возможность достаточно быстрого ускорения при появлении доступной пропускной способности и поддержания скорости передачи данных, близкой к оптимальной при незначительной заполненности очереди маршрутизатора.

2. Аспекты реализации TCP TIPS

Рассматривая аспекты реализации протокола TCP TIPS, необходимо затронуть два момента: совместимость с TCP и техническую сторону реализации алгоритмов, используемых в TCP TIPS. Обеспечение совместимости TCP TIPS с TCP возможно при сохранении формата заголовков сегментов данных, используемого в TCP, а также применении аналогичной техники установки соединения («тройное рукопожатие»).

Так как TCP TIPS нуждается в двух дополнительных полях – TI и PS, – то расширение заголовка TCP можно провести путем использования опций TCP аналогично тому, как это предложено в [7]. Такой подход сохраняет совместимость с заголовком TCP и добавляет в конец заголовка две 32-битные опции (PS – для сегментов с данными, TI – для сегментов с подтверждениями). Использование микросекунд в качестве единиц измерения интервалов времени для поля TI дает возможность различать интервалы при скорости до 64 Гбит/с при размере пакетов 8192 байта. Для достижения больших скоростей можно реализовать масштабирование для значений поля TI, например, представив это поле как 31-битное значение интервала и 1-битное значение, определяющее единицу измерения. Если определить 0 — мкс, а 1 — нс, то полученное 31-битное поле для данных сможет вместить интервалы от 1 нс до 2^{31} мкс (более 35 минут).

Альтернативным способом организации полей TI и PS может быть реализация их в виде единой опции фиксированного или переменного размера. Переменный размер позволяет уменьшить размер заголовка, необходимого TCP TIPS, а опция фиксированного размера дает возможность иметь заголовок постоянной величины и использовать механизмы, полагающиеся на это, например, PL-PMTUD [8].

Рассматривая аспекты реализации TCP TIPS в рамках сетевой подсистемы какой-либо из ОС, необходимо в первую очередь обратить внимание на вопросы, связанные с вычислением времени. Эти вопросы касаются вычисления значений межсегментных интервалов на стороне получателя и реализации высокоточной диспетчеризации сегментов на стороне отправителя. Первый вопрос достаточно подробно

освещен в [8]. Для решения второй задачи необходима поддержка таймеров высокого разрешения. Примером таймеров высокого разрешения являются hrtimers [9], реализованные в ОС Linux.

Несмотря на высокую частоту часов, лежащих в основе таймера, обработка события может быть отложена на некоторый срок, например, из-за выполнения других программных прерываний. TCP TIPS требует отправки данных через межсегментные интервалы, которые могут быть достаточно малы, поэтому для реализации TCP TIPS в ОС может требоваться гарантированное время обработки прерывания. Такие гарантии может дать только операционная система реального времени. Также следует иметь в виду, что необходимость частой обработки прерываний для отправки данных может потребовать больших расходов производительности.

В связи с необходимостью точного вычисления интервалов времени получателем и точного соблюдения интервалов времени отправителем, следует рассмотреть еще один вопрос. В модели отправку данных после их передачи сетевому уровню можно считать мгновенной. В реальных ОС следует учитывать время, затрачиваемое на передачу данных от транспортного уровня до физического, где происходит их фактическая отправка. Если транспортный и сетевой уровни реализуются обычно ядром ОС, то уровень передачи данных и физический уровень – драйвером сетевого оборудования. Разные драйверы могут иметь различные задержки. Если каждый сегмент передается драйвером ядру отдельно, то, при условии, что время задержки при передаче/приеме варьируется от сегмента к сегменту незначительно, значения межсегментных интервалов, используемых отправителем и наблюдаемых получателем, искажаться не будут.

Сетевое оборудование может применять различные оптимизационные техники, такие как LRO и GSO/TSO, для снижения нагрузки на процессор. Аналогичная функциональность может быть представлена и программно. Примером такой реализации является программный интерфейс NAPI в Linux. В таких условиях рассчитывать значение межсегментного интервала или поддерживать отправку сегментов через нужные промежутки времени на транспортном уровне невозможно. Однако необходимо отметить, что TCP TIPS позволяет распределить алгоритм управления потоком между транспортным уровнем сетевой подсистемы ОС и аппаратным уровнем (реализацией драйвера или сетевого оборудования) аналогично тому, как это происходит в случае использования LRO и GSO/TSO. Получатель TCP TIPS может рассчитывать интервалы времени на основе значений, вычисленных драйвером сетевого оборудования (или аппаратно). Более того, сетевое оборудование в состоянии самостоятельно рассчитывать значение поля TI, так как это не требует реализации сложной логики.

Со стороны отправителя следует заметить, что изменение значения межсегментного интервала, используемого отправителем, а также вычисление среднего значения межсегментного интервала, наблюдаемого получателем, происходят один раз за сравнительно большой промежуток времени, равный RTT. Аппаратная составляющая реализации могла бы подсчитывать среднее арифметическое значение поля TI, обеспечивать отправку сегментов через указанное ей значение интервала времени и вести сквозную нумерацию отправляемых сегментов (поле PS). В этом случае программная реализация отправителя TCP TIPS сводилась бы к получению раз

в RTT значения межсегментного интервала, наблюдаемого получателем, расчету нового значения межсегментного интервала, используемого отправителем, согласно алгоритму управления потоком данных, и передаче этого значения аппаратной части реализации. Данные, передаваемые транспортным уровнем аппаратной части реализации, в этом случае ограничиваются окном получателя и алгоритмом Нэйгла и его аналогами. Никакой модификации сетевой подсистемы ОС для обеспечения диспетчеризации сегментов не требуется, эта задача решается аппаратной частью реализации.

ОС может накладывать и другие ограничения на реализацию TCP TIPS. Например, в ОС Linux не рекомендуется использовать операции над числами с плавающей запятой. Это ограничение не является критическим для TCP TIPS, так как используемые в нем алгоритмы хорошо приспособлены для работы с числами с фиксированной точностью.

Можно заключить, что, несмотря на наличие сложностей, связанных с применением высокоточных источников времени и таймеров, реализация TCP TIPS в операционных системах возможна и все инструменты для создания ее компонентов существуют. Что касается применимости TCP TIPS в высокоскоростных сценариях, когда для увеличения производительности приходится использовать аппаратные техники, такие как LRO и GSO/TSO, то протокол TCP TIPS обладает хорошим потенциалом к аппаратной реализации ресурсоемких частей, связанных с расчетом межсегментных интервалов и диспетчеризацией сегментов. При условии аппаратной реализации этих составляющих алгоритма для TCP TIPS также возможно применение таких техник, как LRO и GSO/TSO.

Список литературы

1. Allman M., Paxson V., Blanton E., TCP Congestion Control // RFC 5681, 2009.

2. Сивов А. А. О росте задержек при передаче данных в коммуникационных сетях // Всероссийский конкурс научно-исследовательских работ студентов и аспирантов в области информатики и информационных технологий: сб. науч. работ в 3 т. Белгород, 2012. Т. 2. С. 292–295.

3. Brakmo L., O'Malley S., Peterson L. TCP Vegas: New Techniques for Congestion Detection and Avoidance. ACM SIGCOMM. P. 24–35.

4. Alekseev I.V., Sokolov V.A., Modeling and Traffic Analysis of the Adaptive Rate Transport Protocol // Future Generation Computer Systems. North-Holland: ELSVIER, 2002. V. 18, № 6. P. 813–827.

5. Сивов А. А. Проактивная схема борьбы с перегрузкой в транспортном протоколе на основе оценки доступной пропускной способности сети: материалы VIII Международной научно-практической конференции «Современное состояние естественных и технических наук». М.: Спутник +, 2012. С. 108–118.

6. Сивов А. А. Проактивная схема борьбы с перегрузкой в сети для транспортного протокола // III Международная заочная научно-практическая конференция «Научная дискуссия: вопросы физики, математики, информатики». М.: Международный центр науки и образования, 2012. С. 116–128.

7. Сивов А. А. Формат пакета ARTCP. Особенности формирования и обработки заголовков ARTCP в сетевой подсистеме ОС Linux 2.6 // Моделирование и анализ информационных систем. 2011. Том 18, № 2. С. 129–138.

8. Mathis M., Heffner J. Packetization Layer Path MTU Discovery // RFC 4821. 2007.

9. Gleixner T., Niehaus D. Hrtimers and Beyond: Transforming the Linux Time Subsystems // Proceedings of the Linux Symposium. 2006. Vol. 1. P. 333–346.

TCP TIPS: Transport Protocol for Unreliable Networks with Latency-Sensitive Data

Sivov A.A.

Keywords: TCP TIPS protocol, transport protocols, TCP, congestion avoidance, network simulation, ns-2

The article describes TCP TIPS, the transport level protocol, which aim is to efficiently use the available bandwidth. This protocol implements the proactive methods for congestion avoidance. These methods allow TCP TIPS to yield the required amount of bandwidth to high-priority flows with realtime data and to use this amount, when it becomes available.

Сведения об авторе:
Сивов Анатолий Александрович,
Ярославский государственный университет им. П.Г. Демидова,
аспирант